

**Compte rendu TP B1 TP06 : La société**  
**TechCorp**

**Sommaire**

|                   |   |
|-------------------|---|
| Introduction..... | 1 |
| PHASE 1.....      | 1 |
| Exercice 1.....   | 1 |
| Exercice 2.....   | 2 |
| Exercice 3.....   | 2 |
| Exercice 4.....   | 2 |
| Exercice 5.....   | 2 |
| Exercice 6.....   | 2 |
| Exercice 7.....   | 3 |
| Exercice 8.....   | 3 |
| Exercice 9.....   | 4 |
| Exercice 10.....  | 4 |
| Exercice 11.....  | 5 |
| Exercice 12.....  | 5 |
| PHASE 2.....      | 6 |
| Exercice 13.....  | 6 |
| Exercice 14.....  | 6 |
| Exercice 15.....  | 6 |
| Exercice 16.....  | 7 |
| Exercice 17.....  | 7 |
| Exercice 18.....  | 7 |
| Exercice 19.....  | 7 |
| Exercice 20.....  | 7 |
| Exercice 21.....  | 8 |
| Exercice 22.....  | 8 |

|                        |           |
|------------------------|-----------|
| Exercice 23.....       | 8         |
| Exercice 24.....       | 8         |
| Exercice 25.....       | 9         |
| Exercice 26.....       | 9         |
| Exercice 27.....       | 9         |
| <b>PHASE 3.....</b>    | <b>9</b>  |
| Exercice 28.....       | 9         |
| Exercice 29.....       | 10        |
| Exercice 30.....       | 10        |
| Exercice 31.....       | 10        |
| Exercice 32.....       | 10        |
| Exercice 33.....       | 10        |
| Exercice 34.....       | 11        |
| Exercice 35.....       | 11        |
| Exercice 36.....       | 12        |
| Exercice 37.....       | 12        |
| Exercice 38.....       | 12        |
| Exercice 39.....       | 12        |
| Exercice 40.....       | 14        |
| Exercice 41.....       | 14        |
| <b>PHASE 4.....</b>    | <b>14</b> |
| Exercice 42.....       | 14        |
| Exercice 43.....       | 15        |
| Exercice 44.....       | 15        |
| Exercice 45.....       | 15        |
| Exercice 46.....       | 15        |
| Exercice 47.....       | 15        |
| Exercice 48.....       | 16        |
| Exercice 49.....       | 16        |
| Exercice 50.....       | 17        |
| <b>Conclusion.....</b> | <b>17</b> |

# **Introduction**

Ce TP m'a permis de découvrir la ligne de commande et d'apprendre à dialoguer directement avec l'ordinateur à travers des commandes simples. J'ai compris comment naviguer entre les dossiers, créer, déplacer ou supprimer des fichiers, et utiliser quelques outils système. C'était l'occasion de mieux saisir le fonctionnement du système d'exploitation et l'utilité des commandes au quotidien.

## **PHASE 1**

### **Exercice 1**

La ligne de commande s'appelle aussi interface en ligne de commande (CLI). C'est une interface textuelle où on tape des commandes au clavier pour parler directement à l'ordinateur.

### **Exercice 2**

Pour basculer sur le disque D j'ai taper la commande ci dessous

```
C:\Users\ldv>D:
```

### **Exercice 3**

Pour afficher tous les lecteurs logiques j'ai tapé :

```
wmic logicaldisk get name
```

et j'ai obtenue ceci :

```
Name  
C:  
D:
```

### **Exercice 4**

```
C:\Users\ldv>cd "D:\Dossiers Partagés"
```

Les guillemets sont nécessaires car le chemin contient un espace ("Dossiers Partagés"). Sans guillemets, la commande serait mal interprétée et ne fonctionnerait pas correctement.

## **Exercice 5**

Un chemin absolu correspond à l'adresse complète d'un fichier ou dossier, en partant de la racine du disque (par exemple, `D:\Dossiers Partagés`). Il permet d'accéder directement à l'emplacement sans tenir compte du répertoire courant.

Un chemin relatif indique l'emplacement d'un fichier ou dossier par rapport au répertoire actuel dans lequel on se trouve (par exemple, `Dossiers Partagés` ou `..\Documents`). Il dépend donc du dossier de travail courant.

## **Exercice 6**

`dir /?` (affiche l'aide complète de la commande `dir`)

Terme formel : Ce sont les paramètres ou options (parfois appelés "switches") qui modifient le comportement d'une commande (ex : `/O D`).

## **Exercice 7**

La commande `ls` affiche les fichiers et dossiers présents.

Signification de `.` et `..` :

- `.` représente le répertoire courant (le dossier où l'on se trouve).

- `..` représente le répertoire parent (le dossier au-dessus dans l'arborescence).

## Exercice 8

Pour trier la liste des fichiers et dossiers par date de création sous Windows, on utilise la commande suivante.

```
C:\Users\ldv>dir /T:C /O:D
Le volume dans le lecteur C n'a pas de nom.
Le numéro de série du volume est 504E-4571

Répertoire de C:\Users\ldv

29/08/2025  14:06    <DIR>          ..
29/08/2025  14:06    <DIR>          .
29/08/2025  14:06    <DIR>        Saved Games
29/08/2025  14:06    <DIR>        Pictures
29/08/2025  14:06    <DIR>        Music
29/08/2025  14:06    <DIR>        Links
29/08/2025  14:06    <DIR>        Favorites
29/08/2025  14:06    <DIR>        Downloads
29/08/2025  14:06    <DIR>        Documents
29/08/2025  14:06    <DIR>        Desktop
29/08/2025  14:06    <DIR>        Videos
29/08/2025  14:07    <DIR>        3D Objects
29/08/2025  14:07    <DIR>        Contacts
29/08/2025  14:07    <DIR>        Searches
15/09/2025  14:17    <DIR>        rom
18/09/2025  09:55             15 advanced_ip_scanner_Aliases.bin
18/09/2025  09:55             15 advanced_ip_scanner_Comments.bin
18/09/2025  09:55          332 advanced_ip_scanner_MAC.bin
29/09/2025  13:22             12 123
29/09/2025  13:23              0 1234
29/09/2025  13:25             21 123.txt
29/09/2025  13:25             16 1234.txt
01/10/2025  13:54    <DIR>        .vscode
              7 fichier(s)              411 octets
              16 Rép(s)  136 023 638 016 octets libres
```

**/T:C** : utilise la date de création des fichiers/dossiers comme critère de tri.

**/O:D** : trie les éléments par date, du plus ancien au plus récent.

## Exercice 9

Les deux raccourcis clavier pour se déplacer rapidement dans une ligne de commande sont :

**Home** : place le curseur au début de la ligne.

**End** : place le curseur à la fin de la ligne.

## Exercice 10

Le raccourci clavier qui permet de déplacer le curseur d'un mot à la fois dépend du système d'exploitation :

- Sur Windows/Linux :  
Ctrl + Flèche gauche / Flèche droite
- Sur macOS :  
Option (⌘) + Flèche gauche / Flèche droite

## Exercice 11

Pour afficher l' historique des commandes dans le cmd j'ai taper :  
doskey /history

## Exercice 12

Pour afficher la structure complète (arborescence) du lecteur D: sous Windows, j'ai utilisé la commande suivante.

```
C:\Users\ldv>tree D: /F /A
Structure du dossier
Le numéro de série du volume est 2C4C-0FB2
D:\
| 1412.txt
| 2306
| DRIVERQUERY
| FC
|
+---romain
| 02012008
| 123.txt
| 1234.txt
| 141207.txt
| FC
|
\---test
| DRIVERQUERY
```

**tree** : commande qui affiche l'arborescence des dossiers.

**D :** spécifie le lecteur à explorer.

**/F** : affiche les fichiers en plus des dossiers.

**/A** : utilise des caractères ASCII simples (utile pour une meilleure compatibilité, par exemple si vous copiez-collez).

## **PHASE 2**

### **Exercice 13**

Pour créer deux répertoires sur le lecteur D: en une seule commande, en tenant compte des espaces dans les noms, il faut utiliser des guillemets pour bien gérer les noms avec ces espaces.

Voici la commande que j'ai utilisé dans l'invite de commandes:

```
C:\Users\ldv>mkdir "D:\Archives 2024" "D:\Notes du Personnel"
```

### **Exercice 14**

Dans l'invite de commandes Windows, pour afficher les fichiers y compris les fichiers cachés, je me suis servi de:

```
C:\Users\ldv>dir /a
Le volume dans le lecteur C n'a pas de nom.
Le numéro de série du volume est 504E-4571

Répertoire de C:\Users\ldv

02/10/2025  11:56    <DIR>          .
02/10/2025  11:56    <DIR>          ..
01/10/2025  13:54    <DIR>          .vscode
09/09/2025  13:22             12 123
09/09/2025  13:25             21 123.txt
09/09/2025  13:23              0 1234
09/09/2025  13:25             16 1234.txt
```

`dir` liste les fichiers et dossiers

`/a` affiche tous les fichiers, y compris les cachés et les fichiers système

## Exercice 15

,voici la commande que j'ai entré pour créer le fichier `rapport.txt` dans le dossier Archives 2024 avec le texte :

```
C:\Users\ldv>echo Rapport d'audit initial. > "Archives 2024\rapport.txt"
```

## Exercice 16

Pour ajouter la ligne "Fermeture de l'audit." à un fichier sans écraser son contenu précédent, j'ai utiliser la commande `echo` avec la redirection `>>`.

## Exercice 17

Pour créer un fichier vide nommé **cache.tmp** dans le répertoire **Notes** du dossier **Personnel** en utilisant la commande `echo`, j'ai utiliser la commande suivante :

```
C:\Users\ldv>echo. > Personnel\Notes\cache.tmp
```

`echo.` : affiche une ligne vide.

`>` : redirige la sortie vers le fichier (créé s'il n'existe pas, écrasé s'il existe).

`Personnel\Notes\cache.tmp` : chemin relatif du fichier à créer.

## Exercice 18

Pour afficher le contenu final du fichier `rapport.txt` directement dans l'interface de commande je me suis servis de la commande

```
C:\Users\ldv>type rapport.txt
```

## Exercice 19

Voici la commande pour dupliquer un fichier `rapport.txt` dans le même dossier en le renommant `rapport_v2.txt` :

```
C:\Users\ldv>copy "Archives 2024\rapport.txt" "Archives 2024\rapport_v2.txt"
```



`copy` : commande pour copier un fichier.

"Archives 2024\rapport.txt" : chemin du fichier source.

"Archives 2024\rapport\_v2.txt" : chemin et nom du fichier de destination.

## Exercice 20

Pour déplacer le fichier `rapport_v2.txt` du répertoire actuel (`D:\Archives 2024`) vers son répertoire **parent immédiat**, J'ai saisi dans l'invite de commande:

```
C:\Users\ldv>move rapport v2.txt ..
```

`rapport_v2.txt` : le fichier à déplacer.

`..` : représente le répertoire parent du répertoire courant.

## Exercice 21

Pour renommer le fichier `rapport_v2.txt` situé dans le répertoire parent en `rapport_final.doc`, la commande est la suivante :

```
C:\Users\ldv>mv ../rapport_v2.txt ../rapport_final.doc
```

`mv` : commande pour déplacer ou renommer un fichier.

`../rapport_v2.txt` : chemin vers le fichier dans le répertoire parent.

`../rapport_final.doc` : nouveau nom (et même emplacement) pour le fichier.

## Exercice 22

Si le répertoire contient des fichiers, la commande `rmdir` sans paramètre échoue et affiche une erreur, car elle ne peut supprimer que des répertoires vides.

## Exercice 23

Pour Suppression de Dossier Non Vide, j'ai taper cette commande.

```
C:\Users\eleve>rm -r nom_du_dossier
```

Le dossier `nom_du_dossier` ainsi que tous les fichiers et sous-dossiers qu'il contient seront supprimés.

Aucun message ne s'affiche si la suppression réussit.

Si un fichier est en lecture seule, la commande peut demander une confirmation, sauf si on ajoute `-f` (force).

## Exercice 24

La commande `del` ne fonctionne pas sur un répertoire car elle est conçue pour supprimer uniquement des fichiers. Les répertoires ne peuvent pas être supprimés avec `del` car ils peuvent contenir d'autres fichiers ou dossiers, et leur suppression nécessite une commande spécifique qui gère la suppression récursive, comme `rmdir` ou `rm -r`. Utiliser `del` sur un répertoire génère donc une erreur.

## Exercice 25

J'ai utilisé la commande ci dessous

```
C:\Users\eleve>del temp_*.doc
```

Commande : `del` (delete)

Caractère générique : `*` également

## Exercice 26

Pour voir les différences entre les 2 fichiers, j'ai tapé cette commande

```
C:\Users\eleve>diff spec_old.txt spec_new.txt
```

La commande `diff` compare ligne par ligne le contenu de deux fichiers texte (ici `spec_old.txt` et `spec_new.txt`) et affiche les différences entre eux.

## Exercice 27

Pour déplacer `rapport.txt` du repertoire `D:\Archives 2024` vers `D:\Notes du Personnel` j'ai tapé la commande ci dessous

```
C:\Users\eleve>move "D:\Archives 2024\rapport.txt" "D:\Notes du Personnel"
```

Les guillemets (") sont nécessaires car les noms de dossiers contiennent des espaces.

La commande `move` est utilisée dans l'invite de commandes Windows (cmd) pour déplacer des fichiers.

## PHASE 3

### Exercice 28

En tapant le nom de l'exécutable

En utilisant un chemin relatif ou absolu

En passant par un interpréteur (ex. : bash, python)

### Exercice 29

Pour qu'un programme puisse être lancé par invocation directe, son exécutable doit se trouver dans un répertoire présent dans la variable d'environnement **PATH**. Ainsi, le système peut le localiser et l'exécuter simplement en tapant son nom.

### Exercice 30

La variable d'environnement **PATH** contient la liste des répertoires dans lesquels le système recherche les fichiers exécutables. Lors du lancement d'une application, le système parcourt ces répertoires pour trouver et exécuter le programme demandé sans avoir à spécifier son chemin complet.

### Exercice 31

Pour Afficher la variable d'environnement PATH de Windows j'ai taper la commande ci dessous

```
C:\Users\eleve>echo %PATH%  
C:\Program Files (x86)\Common Files\Oracle\Java  
v1.0\;C:\WINDOWS\System32\OpenSSH\;C:\Program F
```

## Exercice 32

Pour lancer l'application avec start j'ai tapé la commande ci dessous

```
C:\Users\eleve>start "" "%USERPROFILE%\Bureau\Rapport Urgence.lnk"
```

`start` est la commande pour lancer une application ou un fichier.

`""` : un titre de fenêtre vide est nécessaire si le chemin contient des espaces.

`%USERPROFILE%\Bureau\Rapport Urgence.lnk` : chemin complet vers le raccourci sur le bureau de l'utilisateur actuel.

## Exercice 33

La commande `start` prend le premier argument entre guillemets comme titre de la fenêtre. Pour lancer un programme dont le chemin contient des espaces, il faut fournir un titre vide `""` en premier paramètre afin que le chemin soit correctement interprété et le programme lancé.

## Exercice 34

La commande utilisée pour définir la valeur des variables d'environnement, comme par exemple pour modifier ou ajouter un chemin dans la variable PATH, est :

`export`

Exemple pour ajouter un chemin dans le **PATH** :

`export PATH=$PATH:/nouveau/chemin`

La commande `set` sert plutôt à définir ou afficher des variables, mais ne met pas toujours les variables en tant que variables d'environnement accessibles aux processus enfants.

## Exercice 35

J'ai tapé la commande ci dessous

```
C:\Users\eleve>mkdir TestDir && echo Success  
Success
```

`mkdir TestDir` crée le répertoire `TestDir`.

Si cette création réussit, `echo Success` s'exécute et affiche "Success".

Si `mkdir` échoue (par exemple si le répertoire existe déjà), `echo Success` ne s'exécutera pas.

## Exercice 36

J'ai tapé la commande ci dessous

```
C:\Users\eleve>dir non_existent_file || echo Erreur
```

`dir non_existent_file` essaie d'afficher un fichier qui n'existe pas → cette commande échoue.

`||` signifie *exécuter la commande suivante uniquement si la précédente a échoué*.  
`echo Erreur` s'exécute donc seulement si la commande `dir` échoue.

## Exercice 37

J'ai tapé la commande ci dessous

```
C:\Users\eleve>commande1; echo Final Check
```

`commande1` : c'est la première commande que je veux exécuter. Elle peut réussir ou échouer.

`;` : c'est un séparateur qui indique que la commande suivante doit s'exécuter quelles que soient les conditions (succès ou échec) de la commande précédente.

`echo Final Check` : c'est la deuxième commande qui s'exécutera toujours, même si `commande1` échoue.

## Exercice 38

J'ai tapé la commande ci dessous

```
C:\Users\eleve>gpg --output Archives2024.tar.gz --decrypt Archives2024.tar.gz.gpg
```

`gpg` : lance l'outil de chiffrement/déchiffrement GnuPG.

`--output Archives2024.tar.gz` : indique le nom du fichier de sortie (le fichier déchiffré).

`--decrypt Archives2024.tar.gz.gpg` : demande de déchiffrer le fichier

`Archives2024.tar.gz.gpg`.

## Exercice 39

J'ai tapé la commande ci dessous

```

C:\Users\eleve>tree | more
Structure du dossier
Le numéro de série du volume est 3403-40E5
C:.
+---.eclipse
a   +---org.eclipse.oomph.p2
a   a   +---cache
a   +---org.eclipse.oomph.setup
a   a   +---cache
a   a   +---setups
a   +---org.eclipse.tips.state
a   +---org.eclipse.tips.ide.state
+---.freemind
a   +---osm
+---.p2
a   +---org.eclipse.equinox.p2.engine
a   +---profileRegistry
+---.vscode
a   +---cli
a   +---extensions
a   +---lzaruba.spock-test-runner-vscode-0.0.3
a   a   +---coverage
a   a   a   +---lcov-report
a   a   a   a   +---src
a   a   a   a   +---services
a   a   a   +---src
a   a   a   +---services
a   a   +---images
a   a   +---out
a   a   a   +---services
-- Suite -- |

```

| est un pipe, qui redirige la sortie de **tree** vers la commande suivante.

**more** est plus basique, affiche page par page (et j'appuie sur espace pour passer à la page suivante).

## Exercice 40

J'ai tapé la commande ci dessous

```
C:\Users\eleve>dir | clip
```

La commande **dir** affiche la liste des fichiers et dossiers.

Le symbole | (pipe) est utilisé pour rediriger la sortie de la première commande vers l'entrée de la deuxième.

La commande **clip** permet de copier l'entrée reçue dans le presse-papiers.

## Exercice 41

Le premier chiffre/lettre dans la commande **color** représente la couleur de fond (background) de la console ou de l'interface.

Pour rappel, dans la commande **color XY** :

- X = couleur de fond
- Y = couleur du texte (avant-plan)

Chaque chiffre ou lettre correspond à une couleur spécifique selon la convention standard.

## PHASE 4

### Exercice 42

J'ai tapé la commande ci dessous

```
C:\Users\eleve>systeminfo

Nom d'hôte: PC-F20X-X
Nom du système d'exploitation: Microsoft Windows 11 Professionnel Éducation
Version du système d'exploitation: 10.0.26100 N/A build 26100
Fabricant du système d'exploitation: Microsoft Corporation
Configuration du système d'exploitation: Station de travail autonome
Type de build du système d'exploitation: Multiprocessor Free
Propriétaire enregistré: eleve
```

### Exercice 43

J'ai tapé la commande ci dessous

```
C:\Users\eleve>driverquery /v
```

Cette commande liste tous les pilotes installés avec des informations supplémentaires comme le type, le nom, la date, etc.

### Exercice 44

J'ai tapé la commande ci dessous

```
C:\Users\eleve>tasklist /svc
```

`tasklist` affiche la liste des processus en cours.

Le paramètre `/svc` affiche les services hébergés dans chaque processus.

### Exercice 45

J'ai tapé la commande ci dessous

```
C:\Users\eleve>taskkill /IM notepad.exe /F
```

`/IM` : spécifie le nom de l'image (processus) à tuer.

`notepad.exe` : nom du processus à arrêter.

`/F` : force l'arrêt du processus.

## Exercice 46

La combinaison de touches alternative est *\*Ctrl + \** (Ctrl + backslash).

## Exercice 47

J'ai tapé la commande ci dessous

```
C:\Users\eleve>ping www.google.com

Envoi d'une requête 'ping' sur www.google.com [172.217.20.164] avec 32 octets de données :
Réponse de 172.217.20.164 : octets=32 temps=35 ms TTL=115
Réponse de 172.217.20.164 : octets=32 temps=35 ms TTL=115
Réponse de 172.217.20.164 : octets=32 temps=53 ms TTL=115
Réponse de 172.217.20.164 : octets=32 temps=38 ms TTL=115

Statistiques Ping pour 172.217.20.164:
    Paquets : envoyés = 4, reçus = 4, perdus = 0 (perte 0%),
Durée approximative des boucles en millisecondes :
    Minimum = 35ms, Maximum = 53ms, Moyenne = 40ms
```

Cela envoie des paquets ICMP vers Google et affiche les résultats, ce qui permet de vérifier si l'ordinateur peut atteindre Google.

## Exercice 48

J'ai tapé la commande ci dessous

```
C:\Users\eleve>tracert www.microsoft.com

Détermination de l'itinéraire vers e13678.dscb.akamaiedge.net [2.20.170.89]
avec un maximum de 30 sauts :

 1  <1 ms    <1 ms    <1 ms    172.31.255.254
 2   1 ms     1 ms     1 ms    routeuradslsio.btsinfogap.org [172.16.255.252]
 3  33 ms    32 ms    32 ms    radius.lnput657.lnput657.rbcj.orange.net [80.10.115.249]
 4  33 ms    32 ms    32 ms    10.123.205.58
 5   *        *        *        Délai d'attente de la demande dépassé.
 6   *        *        *        Délai d'attente de la demande dépassé.
 7   *       73 ms    33 ms    193.251.131.0
 8  40 ms    38 ms    37 ms    81.52.187.82
 9   *        *        *        Délai d'attente de la demande dépassé.
10   *        *        *        Délai d'attente de la demande dépassé.
11   *        *        *        Délai d'attente de la demande dépassé.
12  35 ms    35 ms    35 ms    a2-20-170-89.deploy.static.akamaitechnologies.com [2.20.170.89]

Itinéraire déterminé.
```

Ces commandes affichent la liste des routeurs (hops) que le paquet traverse, avec le temps de réponse pour chaque saut, permettant ainsi d'analyser le chemin réseau jusqu'au serveur cible.

## Exercice 49

Pour trouver l'IP d'un nom de domaine, on utilise **nslookup** ou **dig**, par exemple :

nslookup example.com

Pour faire l'inverse (trouver le nom à partir d'une IP), on fait :



nslookup 192.0.2.1

## **Exercice 50**

Pour redémarrer un ordinateur Windows et lancer immédiatement le menu des options de démarrage avancées, j'ai utilisé la commande **shutdown** suivante:

```
shutdown /r /o /f /t 0
```

**/r** : redémarrer l'ordinateur

**/o** : aller au menu des options de démarrage avancées (nécessite **/r**)

**/f** : forcer la fermeture des applications ouvertes sans avertissement

**/t 0** : délai de 0 seconde avant le redémarrage

## **Conclusion**

Grâce à ce TP, j'ai pris confiance avec l'invite de commandes et je vois à quel point elle peut être pratique et rapide. J'ai appris à mieux organiser mes fichiers et à comprendre le rôle de certaines commandes système. C'est une expérience utile qui me servira sûrement dans mes futures manipulations informatiques.